

IN THE CLAIMS:

1 1. (CANCELLED)

1 2. (PREVIOUSLY PRESENTED) The method of claim 6 further comprising the step of
2 defining a synchronization database having one or more synchronization records at the
3 active supervisor, wherein

4 the synchronization records store state information to be transmitted to the
5 standby supervisor.

1 3. (ORIGINAL) The method of claim 2 further comprising the step of updating one or
2 more of the synchronization records in response to an operating change at the at least one
3 application program.

1 4. (ORIGINAL) The method of claim 3 wherein the transmitting step comprises the step
2 of sending the one or more updated synchronization records to the standby supervisor.

1 5. (CANCELLED)

1 6. (PREVIOUSLY PRESENTED) In an intermediate network device having at least
2 one line card defining a plurality of ports for receiving and forwarding messages and two
3 or more supervisors, each supervisor configured to run one or more applications to facili-
4 tate message handling by the network device, a method for continuing operation of at
5 least one application despite a crash or failure, the method comprising the steps of:

6 designating a first supervisor to be an active supervisor and a second supervisor to
7 be a standby supervisor for the network device;

8 executing the at least one application at the active supervisor;

9 holding the at least one application at the standby supervisor in a dormant state;

10 transmitting state information generated during execution of the at least one ap-
11 plication from the active supervisor to the standby supervisor;

12 storing the state information at the standby supervisor;

13 in response to the standby supervisor detecting a crash or failure at the active su-
14 pervisor, carrying on execution of the at least one application at the standby supervisor
15 based upon at least some of the stored state information;

16 generating a sequence number for use in instructing the at least one line card to
17 change operating condition;

18 sending the sequence number to the at least one line card with the instruction;

19 storing the sequence number at the at least one line card;

20 sending the sequence number to the standby supervisor;

21 storing the sequence number at the standby supervisor; and

22 in response to the standby supervisor detecting the crash or failure of the active
23 supervisor, comparing the sequence number stored at the standby supervisor with the se-
24 quence number at the at least one line card.

1 7. (PREVIOUSLY PRESENTED) The method of claim 6 further comprising the step of
2 continuing operation of the at least one line card, following the crash or failure of the ac-
3 tive supervisor, if the sequence number stored at the at least one line card is one of (a)
4 less than or equal to or (b) greater than the sequence number stored at the standby super-
5 visor.

1 8. (PREVIOUSLY PRESENTED) The method of claim 7 further comprising the step of
2 resetting the at least one line card, following the crash or failure of the active supervisor,
3 if the sequence number stored at the at least one line card is one of (a) greater than or (b)
4 less than or equal to the sequence number stored at the standby supervisor.

1 9. (PREVIOUSLY PRESENTED) The method of claim 6 further comprising the steps
2 of:

3 determining the validity of the state information stored at the standby supervisor
4 following the crash or failure of the active supervisor; and

5 blocking the at least one application from utilizing state information determined
6 to be invalid in its execution.

1 10. (PREVIOUSLY PRESENTED) In an intermediate network device having at least
2 one line card defining a plurality of ports for receiving and forwarding messages and two
3 or more supervisors, each supervisor configured to run one or more applications to facili-
4 tate message handling by the network device, a method for continuing operation of at
5 least one application despite a crash or failure, the method comprising the steps of:

6 designating a first supervisor to be an active supervisor and a second supervisor to
7 be a standby supervisor for the network device;

8 executing the at least one application at the active supervisor;

9 holding the at least one application at the standby supervisor in a dormant state;

10 transmitting state information generated during execution of the at least one ap-
11 plication from the active supervisor to the standby supervisor;

12 storing the state information at the standby supervisor;

13 in response to the standby supervisor detecting a crash or failure at the active su-
14 pervisor, carrying on execution of the at least one application at the standby supervisor
15 based upon at least some of the stored state information;

16 creating, at the active supervisor, an instance of an event in response to a request
17 from an application;

18 providing the event instance to the requesting application for processing;

19 providing the event instance to any listening applications that have registered for
20 the event for processing;

21 passing the event instance to the standby supervisor;

22 receiving notifications from the requesting and listening applications that they
23 have completed their processing of the event instance;

24 passing the notifications to the standby supervisor; and

25 in response to the standby supervisor receiving notifications from the requesting
26 and all listening applications, closing the event instance at the active and standby supervi-
27 sors.

1 11. (PREVIOUSLY PRESENTED) The method of claim 10 further comprising the step
2 of:

3 in response to the standby supervisor detecting the crash or failure of the active
4 supervisor, determining whether one or more event instances passed to the standby su-
5 pervisor remain open;

6 for a given event instance that remains open, identifying the requesting and listen-
7 ing applications that have not completed their processing of the given event instance;

8 for each requesting and listening application that has not completed its processing
9 of the given event instance, calling a recovery function defined by the respective applica-
10 tion to handle the open event instance.

1 12. (CANCELLED)

1 13. (CANCELLED)

1 14. (CANCELLED)

1 15. (CANCELLED)

1 16. (PREVIOUSLY PRESENTED) A method for operating a network device, compris-
2 ing:

3 operating an active supervisor, the active supervisor creating an event instance in
4 response to a change in operating state from a requesting application;

5 providing, by the active supervisor, the event instance to the requesting applica-
6 tion and listening applications that have registered for the event for processing;

7 passing, by the active supervisor, the event instance to a standby supervisor;

8 subsequent to the steps of providing and passing, receiving, at the active supervi-
9 sor, notifications from the requesting and listening applications that the requesting and
10 listening applications have completed processing of the event instance;

11 passing, by the active supervisor, the notifications to the standby supervisor; and

12 in response to receiving the notifications from the requesting and all listening ap-
13 plications, closing the event instance at the active and standby supervisors.

1 17. (PREVIOUSLY PRESENTED) The method of claim 16, further comprising:

2 in response to a failure of the active supervisor, determining whether one or more
3 event instances passed to the standby supervisor remain open;

4 identifying the requesting and listening applications, if any, that have not com-
5 pleted their processing of an open event instance; and

6 calling a recovery function defined by the respective application to handle the
7 open event instance.

1 18. (CANCELLED)

1 19. (PREVIOUSLY PRESENTED) A network device, comprising:

2 an active supervisor to run applications, the active supervisor to create an event
3 instance in response to a change in operating state from a requesting application, to pro-
4 vide the event instance to the requesting application and listening applications that have
5 registered for the event for processing, and subsequent to providing the event instance, to
6 receive notifications from the requesting and listening applications that the requesting
7 and listening applications have completed processing of the event instance; and

8 a standby supervisor to receive the event instance and the notifications from the
9 active supervisor, where in response to receiving notifications from the requesting and all
10 listening applications, the active and standby supervisors are to close the event instance.

1 20. (PREVIOUSLY PRESENTED) The network device of claim 19, further compris-
2 ing:

3 in response to the standby supervisor detecting a failure of the active supervisor,
4 the standby supervisor is further to determine whether one or more event instances passed
5 to the standby supervisor remain open, to identify the requesting and listening applica-
6 tions, if any, that have not completed their processing of an open event instance, and to
7 call a recovery function defined by the respective application to handle the open event
8 instance.

1 21. (CANCELLED)

1 22. (PREVIOUSLY PRESENTED) A network device, comprising:

2 means for operating an active supervisor, the active supervisor creating an event
3 instance in response to a change in operating state from a requesting application;

4 means for providing, by the active supervisor, the event instance to the requesting
5 application and any listening applications that have registered for the event for process-
6 ing;

7 means for passing, by the active supervisor, the event instance to a standby super-
8 visor;

9 means for receiving, at the active supervisor, subsequent to the providing and
10 passing, notifications from the requesting and listening applications that the requesting
11 and listening applications have completed processing of the event instance;

12 means for passing, by the active supervisor, the notifications to the standby super-
13 visor; and

14 means for in response to receiving notifications from the requesting and all listen-
15 ing applications, closing the event instance at the active and standby supervisors.

1 23. (PREVIOUSLY PRESENTED) The network device of claim 22, further compris-
2 ing:

3 in response to the standby supervisor detecting a failure of the active supervisor,
4 means for determining whether one or more event instances passed to the standby super-
5 visor remain open;

6 means for identifying the requesting and listening applications, if any, that have
7 not completed their processing of an open event instance; and

8 means for calling a recovery function defined by the respective application to
9 handle the open event instance.

1 24. (CANCELLED)

1 25. (PREVIOUSLY PRESENTED) A computer readable media, comprising: the com-
2 puter readable media containing instructions for execution on a processor for the practice
3 of the method of,

4 operating an active supervisor, the active supervisor creating an event instance in
5 response to a change in operating state from a requesting application;

6 providing, by the active supervisor, the event instance to the requesting applica-
7 tion and listening applications that have registered for the event for processing;

8 passing, by the active supervisor, the event instance to a standby supervisor;

9 subsequent to the steps of providing and passing, receiving, at the active supervi-
10 sor, notifications from the requesting and listening applications that the requesting and
11 listening applications have completed processing of the event instance;

12 passing, by the active supervisor, the notifications to the standby supervisor; and

13 in response to receiving notifications from the requesting and all listening applica-
14 tions, closing the event instance at the active and standby supervisors.

1 26. (PREVIOUSLY PRESENTED) The computer readable media of claim 25, further
2 carrying instructions for the method of,

3 in response to the standby supervisor detecting a failure of the active supervisor,
4 determining whether one or more event instances passed to the standby supervisor remain
5 open;

6 identifying the requesting and listening applications, if any, that have not com-
7 pleted their processing of an open event instance; and

8 calling a recovery function defined by the respective application to handle the
9 open event instance.

1 27. (CANCELLED)

1 28. (PREVIOUSLY PRESENTED) Electromagnetic signals propagating on a computer
2 network, comprising: the electromagnetic signals carrying instructions for execution on a
3 processor for the practice of the method of,

4 operating an active supervisor, the active supervisor creating an event instance in
5 response to a change in operating state from a requesting application;

6 providing, by the active supervisor, the event instance to the requesting applica-
7 tion and listening applications that have registered for the event for processing;

8 passing, by the active supervisor, the event instance to a standby supervisor;

9 subsequent to the steps of providing and passing, receiving, at the active supervi-
10 sor, notifications from the requesting and listening applications that the requesting and
11 listening applications have completed processing of the event instance;

12 passing, by the active supervisor, the notifications to the standby supervisor; and

13 in response to receiving notifications from the requesting and all listening applica-
14 tions, closing the event instance at the active and standby supervisors.

1 29. (PREVIOUSLY PRESENTED) The electromagnetic signals of claim 28, further
2 carrying instructions for the method of,

3 in response to the standby supervisor detecting a failure of the active supervisor,
4 determining whether one or more event instances passed to the standby supervisor remain
5 open;

6 identifying the requesting and listening applications, if any, that have not com-
7 pleted their processing of an open event instance; and

8 calling a recovery function defined by the respective application to handle the
9 open event instance.